(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
28 September 2006 (28.09.2006)

PCT

(10) International Publication Number
WO 2006/101402 A1

(51) International Patent Classification:
H04L 29/06 (2006.01) *H04L 29/08* (2006.01)

(21) International Application Number:
PCT/NO2006/000108

(22) International Filing Date: 21 March 2006 (21.03.2006)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
20051487 21 March 2005 (21.03.2005) NO

(71) Applicant *(for all designated States except US)*: **TELENOR ASA** [NO/NO]; Snarøyveien 30, N-1331 Fornebu (NO).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **DO, Thanh Van** [NO/NO]; Stjernemyrveien 28, N-0673 Oslo (NO). **DO, Thuan Van** [NO/NO]; Haugerudveien 48, N-0674 Oslo (NO). **JØRSTAD, Ivar** [NO/NO]; Bjølsengata 15, N-0468 Oslo (NO).

(74) Agent: **OSLO PATENTKONTOR AS**; P.O. Box 7007m, N-0306 Oslo (NO).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, LY, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

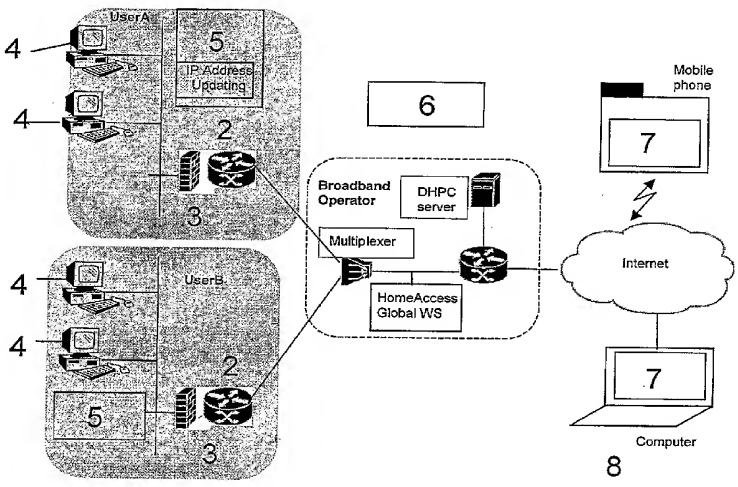
(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Declaration under Rule 4.17:
— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

Published:
— *with international search report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: A METHOD AND DEVICE FOR ACCESSING SERVICES AND FILES



(57) Abstract: The present invention relates to a method and device for accessing services and files on a computer (4) in a home network (1) from a stationary or mobile device (8, 9) outside said local network Said local network is equipped with a networked file system, and said stationary or mobile device is able to communicate with said local network over a wide area network. For any networked file system message that is to be transmitted over said wide area network, at least some fields of said networked file system message are mapped into corresponding fields in an XML message representing said networked system message. Any XML message that is received over said wide area network, said XML message is parsed, and if said XML message represents a networked file system message, a networked file system message is reconstructed by mapping each field of said XML message into a corresponding field of said reconstructed networked file message.

WO 2006/101402 A1

A METHOD AND DEVICE FOR ACCESSING SERVICES AND FILES

Field of the invention

The present invention relates to data communication systems, and in particular the access and use of files and services residing on a local network computer system from remote computers, Personal Digital Assistants (PDA) or mobile phones.

Technical background

Nowadays, more and more households are acquiring broadband Internet connections to their home. On the home local network they may have several computers running various services and hosting private documents and files. It is hence quite desirable to be able to access these services and files from outside their home via stationary computers or mobile devices like cellular phone or PDA.

Unfortunately, to be connected to the Internet does not bring only advantages but also drawbacks. The user home networks as other networks connected to the Internet are exposed to fraudulent attacks and misuses. For protection against frauds and intrusions, more and more users are installing firewalls. While protecting the user home network, the firewalls may also prevent the legitimate users to access their files and services located on the user home network.

In order to establish a secure connection to a local network from outside, solutions such as Virtual Private Networks (VPN) can be used. The firewall of the local network is configured to also operate as a VPN server, and the mobile device could be used as a VPN client to connect to this VPN server. Having connected to this server, the mobile device becomes part of the local network that resides behind the firewall/VPN server. This Virtual

Network (realised with an encrypted tunnel) will allow traffic of any service to flow back and forth between the local network and the mobile device.

The VPN solution has, however, many limitations. The VPN solution requires high resource consumption in terms of processing power and network bandwidth, which is usually not available in mobile devices and wireless networks.

Another drawback is that it takes times to start up a VPN and it does time out when there is no activity. This is not convenient for the mobile user that sporadically accesses his home network.

These limitations call for a simplified solution which can adapt to both user's technical skills and the resource constraints in networks and devices.

Summary of the invention

The present invention provides a solution for accessing services and files residing on a computer in a local network from stationary or mobile devices outside said local network without compromising security.

The invention requires only minimal technical skills to take into use. All that is needed is to download and install an application on the computer where services reside (in the home network), as well as to download and install a client on the device in use.

In addition, the invention allows several (all) computers on the local network to provide services, thus every person having their own computer can access their own personal services from their mobile device.

The scope of the invention appears from the appended patent claims.

In particular, the present invention relates to a method for providing access to services and files on a computer in a local network from a stationary or mobile device outside said local network. Said local network is equipped with a networked file system, and said stationary or mobile device is able to communicate with said local network over a wide area network. For any networked file system message that is to be transmitted over said wide area network, at least some fields of said networked file system message are mapped into corresponding fields in an XML message representing said networked system message. Any XML message that is received over said wide area network, said XML message is parsed, and if said XML message represents a networked file system message, a networked file system message is reconstructed by mapping each field of said XML message into a corresponding field of said reconstructed networked file message.

The invention also relates to a device for providing access to services and files on a computer in a local network, from a stationary or mobile device outside said local network. Said local network is equipped with a networked file system, and said stationary or mobile device is connected to said local network over a wide area network. Said device is adapted to map at least some fields of a networked file system message to be transmitted over said wide area network into corresponding fields in an XML message representing said networked file system message. Said device is further adapted to parse a XML message received over said wide area network, and if said XML message represents a networked file system message to reconstruct a networked file system message by mapping each field of said XML message into a corresponding field of said reconstructed networked file message.

Brief description of the drawings

The invention will now be described in detail in reference to the appended drawings, in which:

Figure 1 illustrates the overall architecture of a mobile home access system according to the present invention,

Figure 2 shows a solution for a local network with dynamic global IP address,

Figure 3 and 4 shows a solution for a local network with permanent or dynamic local IP address,

Figure 5 shows the interfaces of the home access local Web service according to the invention,

Figure 6 is a sequence diagram illustrating how one request from a home access Web service client invokes several requests and responses between the home access local Web service and the file system,

Figure 7 is a sequence diagram showing the messages passing using the authentication interfaces.

Detailed description

Figure 1 depicts a typical home broadband connection to the Internet. The underlying network technologies can be xDSLs or cable TV. As shown in Figure 1, a local network 1 may comprise several computers 4 and devices. It is connected to a broadband router 2, e.g an ADSL terminating Unit Router (ATU-R), which may provide DHCP and NAT (Network Address Translation). A firewall 3 should be installed to protect the network 1 against intruders. Such a firewall 3 can also be incorporated in the broadband router 2 or LAN/WLAN router. The broadband router 2 is its turn connected to a multiplexer, e.g. Digital Subscriber Loop

Access Multiplexer (DSLAM). As shown in Figure 1, both DHCP and NAT functions may also be carried out in the broadband operator network.

The solution according to the present invention will allow access to any files or services residing on computers 4 on a LAN 1 behind a firewall 3, typically a private Local Area Network (LAN). The solution as shown Figure 1 consists of three components:

- A Home Access Local Web Service 5 that is installed on the PC to provide access to files and services
- A Home Access Global Web Service 6 addressable by a global IP address
- A Home Access Web Service Client 7 that is installed on the terminal(s) 8, 9 used to access files and services

Based on the nature of the local network IP address, there are four different configurations:

Case 1: Local network using permanent global IP address

In this case, the Mobile Access to local network solution does require only two components:

- Home Access Local Web Service 5
- Home Access Web Service client 7

The Home Access Web Service client 7 interacts with the Home Access Local Web Service 5 to allow the user to access his files and services on his local network.

Case 2: Local network using dynamic global IP address

In this case, the Mobile Home Access requires all the three components:

- Home Access Local Web Service 5
- Home Access Global Web Service 6
- Home Access Web Service Client 7

The Home Access Local Web Service 5, in addition to the functions as described in case 1 must be equipped with the following functions:

- IP address discovery and updating

The Home Access Web Service Client 7 must be equipped with the following function:

- Home Access Local Web Service discovery

As shown in Figure 2 the Home Access Local Web Service 5 is communicating with the Home Access Global Web Service 6 to update its current IP address. The Home Access Web Service Client 7 can then interact directly with the Home Access Local Web Service 5 to access the files and services on the local network.

Case 3: Local network using permanent local IP address

In this case, illustrated in Fig. 4, the Mobile Home Access requires all the three components as in case 2. However, the Home Access Global Web Service 6 must be located in the broadband operator domain. It has the same interfaces as the previously described Home Access Global Web Service 6, but in addition it provides the same interface for file and service access as the Home Access Local Web Service 5.

To access a file or a service located on his local network 1, the user issues a command to the Home Access Web Service Client 7. The Client 7 will invoke the appropriate method on the Home Access Global Web Service 6, which has a permanent URI. Some of the input parameters should be *subscriber_id* and *password*.

Upon successful authentication, the Home Access Global Web Service 6 will invoke appropriate methods on the Home Access Local Web Service 5 residing on the user's local network 1. It is worth noting that the Home Access Global Web Service 6 must know the local network's IP address and use it to invoke methods on the Home Access Local Web Service 5. The approach for acquiring the Home Access Local Web Service IP address is the same as previously described in case 2.

Case 4: Local network using dynamic local IP address

In this case, also illustrated by the fig. 4 drawing, the solution has a configuration which is similar to the case 3. The Home Access Global Web Service 6 needs to find the current IP address of the local network 1 which is now dynamically allocated.

Functionality of the components

Home Access Local Web Service

The role of the Home Access Local WS 5 is to expose the relevant operations of the native file system on the World Wide Web such a mobile client 8, 9 can use them to access files and services located within the local network 1.

As shown in Figure 4, the Home Access Local Web Service 5 has three interfaces:

- The Native File interface 10

- The Web Service File interface 11
- The Administrative interface 12

In addition to the file access functionality the Home Access Local Web Service 5 has also the functionality to periodically query the IP address of the local network 1 and uploads it to the Home Access Global Web Service 6 or a defined globally accessible storage area.

The Native File interface

At the local network, there could be several users sharing several heterogeneous computers and peripheral devices. It is assumed that the local network is equipped with a networked file system that allows the users to view and to access remote files located on other computers from one computer. Examples of such a networked file system can be Sun Network File System (NFS) [1][2] or Common Internet File System (CIFS) [3]. However, only CIFS will be considered further since it is incorporated in Microsoft Windows that are installed in most private households.

CIFS is a file sharing protocol. Client systems use this protocol to request file access services from server systems over a network. It is based on the Server Message Block (SMB) protocol widely in use by personal computers and workstations running a wide variety of operating systems.

The protocol supports the following features:

- File access
- File and record locking
- Safe caching, read-ahead, and write-behind

- File change notification
- Protocol version negotiation
- Extended attributes
- Distributed replicated virtual volumes
- Server name resolution independence
- Batched requests
- Unicode file names

Although CIFS is independent of the transport layer, the common transports today across this interface is through NBT (NetBIOS over TCP) or across raw TCP connections.

Raw TCP transport

This mode of operation is the simplest, where SMB messages can be sent immediately to port 445 on the server. Based on the response to a TCP connection request to this port, and possibly a reply to the SMB message, either RAW transport can used further or an NBT session can be initiated as described below.

NBT transport

In this mode, additional messages must be sent to gain access to the file server and to initiate a session. Also, a valid NetBIOS name of the file server is needed in the message that requests a new session. The need for NBT support completely relies on the servers that should be supported and whether these are expecting correct NBT semantics.

The Web Service File Interface

Since the goal is to enable file and service access from outside devices, especially mobile phones, which have several limitations, the requirements on the Web Service Interface are as follows:

- It should support both regular computers and limited mobile devices
- It should be capable to adapt itself to the device type
- For mobile phones with physical limitations in terms of storage, processing, small display, etc. the operations/methods should be rich such that only few operations are required to accomplish a task.

To cope with these requirements the Web Service File Interface consists of the following sub-interfaces:

- Authentication Interface
- Administration Interface
- Tunnelling Interface
- Reduced Mapping interface

Before allowing access to home files and services, it is important that the identification, authentication and authorisation are carried out properly. The Web Service File Interface must have an Authentication Interface.

The Tunnelling Interface is more suitable for access from remote personal computer, which has a CIFS client installed. The Reduced Mapping Interface is intended formobile devices with limited capabilities.

Authentication Interface

This interface controls identification, authentication and authorization to shared resources.

IAUTHMustAuthenticate(Challenge) - This method is used to notify the client that it is required to authenticate itself prior to accessing any resources through the service access point. This method can be used as a response to any type of request from an unauthenticated client.

IAUTHAuthenticateRequest(Credentials) - This method is used by the client to request authentication by providing proper credentials.

IAUTHAuthenticateResponse() - This method is used by the service access point to notify the client about the outcome of the authentication process.

The Administrative Interface

Access to administration methods requires successful authentication through the interface described earlier. The administrative interface can be used both from remote clients as well as from clients on the local network which could be an administration application.

The Administrative Interface allows a user to specify:

- What directories on the home computer(s) should be made accessible to remote devices

In addition it allows a System Administrator to configure:

- User accounts

IADMListHosts() - Lists all hosts on the local network

IADMListUsers() - Lists all registered users

IADMListDirectoriesOnHost(String host) - Lists all accessible directories on the specified host

IADMSetAccessRights(URI resource, Int accessrights) - Sets the specified access rights on the specified file or directory

IADMGetUserConfiguration(String user_id) - Retrieves the specified user's configuration

IADMSetUserConfiguration(String user_id, Configuration c) - Sets the specified user's configuration

Each user's access rights to resources and preferences are controlled through two methods (*IADMGetUserConfiguration and IADMSetUserConfiguration*). By defining a generic method which passes the configuration as a parameter to the Home Access Local Web Service, maximum flexibility is achieved, and new features can easily be added later on.

A Configuration contains at least the following definitions:

1. Definition of access rights (readable, writable or both) to specific files and directories

2. For each resource it should be possible to define which component of the resource is available (offset, length etc.)

3. Definition of access rights and format/presentation from specific device or group of devices

4. Definition of access rights from specific IP-addresses, subnets or domains

5. Definition of access rights by specific users and groups of users

Tunnelling Interface

In tunnelling mode, a complete CIFS message is encapsulated in a Simple Object Access Protocol (SOAP) message by the Home Access Local Web Service using binary attachments. At the Web Service client side (on the terminal), the CIFS content is extracted from the SOAP message and exposed through a CIFS server. This way, an ordinary CIFS enabled browser (e.g. Windows Explorer) can be used to access the remote file system.

There are basically two approaches to embedding binary data into SOAP messages.

The first approach is to use the XML CDATA element type for embedding the CIFS message into the XML message. The drawback of this solution is that the data must be base64 encoded to avoid the content conflicting with e.g. the terminating CDATA tag. Using base64 encoding results in an increase in size of 1/3 of the original size. For SMB messages containing only signalling information, this might not be a problem, but for the messages containing file contents it is.

The other approach is to use SOAP with Attachments (SwA) [4] [5] [6], but this is not yet supported by all SOAP platforms. It is however supported with JAX-RPC [7] through SAAJ [8]. SwA will, utilising one of the referenced specifications, be supported by all SOAP platforms in the near future.

Also, the client application exposing the file system is required to authenticate itself towards the service access point before access to the remote file system is granted and the file system can be exposed. Except for this authentication process, all other commands follow the network file system protocol.

The Tunnelling Interface has two methods:

ITUNReqCommand(CIFSAttachment) - Transports a complete request command from client to host with network file system

ITUNResCommand(CIFSAttachment) - Transports a complete response command from host with network file system to client

Reduced Mapping Interface

Every CIFS message can be replaced by a corresponding SOAP message. In theory, each field of a CIFS message could be mapped into an entry of a SOAP message by the Home Access Local Web Service. At the client side (terminal), the SOAP message is parsed and the original CIFS message reconstructed and exposed through a CIFS server. However, such a full mapping scheme introduces a lot of overhead and it is not sure that the mobile device is capable to receive and process all the data that it gets. A reduced mapping scheme is more efficient and has the following advantages:

- Reduces the content of each message
- Reduces the number of total messages
- Reduces the requirements on the mobile device; there is no need for a complete CIFS client on the device, but only a client which supports the specified methods.

Only the most important parts of native network file system messages are mapped to an XML format. In addition, a set of management interfaces that are used between the client and the service access point, are defined.

These interfaces control connection establishment towards shares, as well as maintenance of sessions and teardown of connections.

IACCListResources(URI uri, String pattern, Boolean recursive) - Lists all resources on the specified URI matching the specified pattern. If pattern is left empty, all resources on the specified URI are listed. Setting recursive to true allows this method to be used for searching for specific named resources throughout the entire tree defined by uri.

IACCReadResource(URI uri) - Reads the contents of the specified resource as specified in the user configuration described previously. This method incorporates several methods of the network file system, such as protocol negotiation, session setup etc., see the enclosed example.

IACCWriteResource(URI uri, WriteSpecification ws) - Writes to the specified resource the content specified by ws (e.g. create/offset/append, data, length etc.). This method incorporates several methods of the network file system, such as protocol negotiation, session setup etc., see the enclosed example.

Home Access Global Web Service

The Home Access Global Web Service is required for the three cases:

- Dynamic global IP address
- Permanent local IP address
- Dynamic local IP address.

It is collaborating with several Home Access Local Web Services belonging to different users. It must have a list

of users to serve. Before establishing the connection of a Home Access Local Web Service of a user, sufficiently strong authentication must be carried out.

It has the following functionality:

- Discovery and updating of the local network current IP address
- Relaying the method requests from the Home Access Web Service Client to the Home Access Local Web Service.

It has the following interfaces:

- The Native File interface
- The Web Service File interface
- The IP update interface

The two first interfaces are the same as the ones defined for the Home Access Local Service.

The IP update interface has the following method:

IUpdateIP(user_id, IP address) - To update the IP address of the specified user

IGetCurrentIP(user_id) - Returns the current IP address of the specified user

Home Access Web Service Client

There are two types of Home Access Web Service Client:

- Tunnelling Client
- Reduced Mapping Client

The Tunnelling Client will use the Tunnelling Interface to interact with either the Home Access Local Web Service or the Home Access Global Web Service. This Client is suitable for regular PCs. It incorporates also a CIFS server such that a regular CIFS client like Windows Explorer can be used to access the remote files and services.

The Reduced Mapping Client will use the Reduced Mapping Interface to interact with the Home Access Local Web Service and Home Access Global Web Service. This Client is suitable for mobile devices such as mobile phones or PDA (Personal Digital Assistant). It incorporates also a file browser and a User interface (UI) which are designed for devices with limited display and navigation ability.

Example

Reduced Mapping Interface - Message Reduction

As Fig. 5 displays, the Reduced Mapping Interface decreases the number of messages travelling over the network between the mobile device and the local network.

The previous sections of this document have described the generic interfaces necessary to expose a file system on a LAN behind a router/firewall to remote hosts through an XML Web Service. This example details how this can be done using the Common Internet File System (CIFS) as a network file system on the LAN.

This example will provide XML Schema Definitions (XSDs) for transforming CIFS messages into appropriate SOAP messages.

The namespace for all schemas should be http://www.ongx.org/CIFS2XML.

Common Interfaces

This section defines the interfaces that are shared between all modes of operation.

Authentication Interface

The parameters in the following messages employ the Digest Authentication mechanisms described in RFC2617 [9]. This is for illustrative purposes only and other (stronger) authentication mechanisms could/should be employed. The message exchange described below is illustrated in Figure 7.

IAUTHMustAuthenticate(Challenge)

The XSD for this message is defined in *IAUTHMustAuthenticate.xsd* as (and in this case represents the RFC2617 *WWW-Authenticate* request):

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
  <xs:schema xmlns:xs="http://www.ongx.org/CIFS2XML">
        <xs:element name="IAUTHMustAuthenticate">
              <xs:complexType>
                    <xs:sequence>
                          <xs:element name="realm"
                          type="xs:string"/>
                          <xs:element name="nonce"
                          type="xs:string"/>
                          </xs:sequence>
              </xs:complexType>
        </xs:element>
  </xs:schema>
```

IAUTHAuthenticateRequest(Credentials)

The XSD for this message is defined in *IAUTHAuthenticateRequest.xsd* as (and in this case represents the RFC2617 *Authorisation request*):

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
  <xs:schema xmlns:xs="http://www.ongx.org/CIFS2XML">
        <xs:element name="IAUTHAuthenticateRequest">
              <xs:complexType>
                    <xs:sequence>
                      <xs:element                    name="username"
```

```
type="xs:string"/>
                        <xs:element name="realm" type="xs:string"/>
                        <xs:element name="nonce" type="xs:string"/>
                      <xs:element name="uri" type="xs:anyURI"/>
                      <xs:element name="response" type="xs:string"/>
                          </xs:sequence>
                  </xs:complexType>
            </xs:element>
      </xs:schema>
```

IAUTHAuthenticateResponse()

The XSD for this message is defined in IAUTHAuthenticateResponse.xsd as:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
  <xs:schema xmlns:xs="http://www.ongx.org/CIFS2XML">
        <xs:element name="IAUTHAuthenticateResponse">
              <xs:complexType>
                    <xs:sequence>
                          <xs:element name="result"
                          type="xs:boolean"/>
                          </xs:sequence>
                </xs:complexType>
        </xs:element>
  </xs:schema>
```

Tunnelling Mode

In tunnelling mode, a complete binary CIFS message is attached to a SOAP message. In addition to the binary part, the SOAP header must also be present to denote the type of attachment that is included (i.e., a CIFS message) and its identifier (according to the SOAP 1.2 with Attachments defined by World Wide Web Consortium [3]).

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
      <xs:schema  xmlns:xs="http://www.ongx.org/CIFS2XML">
                  xmlns:ref="http://ws-i.org/profiles/basic/1.1/xsd"
            <xs:element name="SMBMessage">
                  <xs:complexType>
                        <xs:sequence>
                              <xs:element name="message"
                              type="ref:swaRef" use="required"/>
                        </xs:sequence>
                  </xs:complexType>
      </xs:element>
</xs:schema>
```

By making the *SMBMessage* element a *complexType*, additional information can be added later on if needed.

A SOAP message with a CIFS message attached would look like this:

```
MIME-Version: 1.0
Content-Type: Multipart/Related; boundary=MIME_boundary;
type=text/xml;
        start="<access2home.xml@ongx.org>"
Content-Description: SOAP message with SMB message
attached.

--MIME_boundary
Content-Type: text/xml; charset=UTF-8
Content-Transfer-Encoding: 8bit
Content-ID: <access2home.xml@ongx.org>

<?xml version='1.0' ?>
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
xmlns:SMB="http://www.ongx.org/CIFS2XML"
<SOAP-ENV:Body>
      <SMB:SMBMessage>
            <message>cid:SMBMessage.MobileAccess1234@ongx.o
rg</message>
      </SMB:SMBMessage>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

--MIME_boundary
Content-Type: application/octet-stream
Content-Transfer-Encoding: binary
Content-ID: <SMBMessage.MobileAccess1234@ongx.org>

...binary SMB message...
--MIME_boundary—
```

The cid value in the Message element refers to the Content-ID tag in the second MIME boundary, and should be unique for each SOAP message. It might be necessary to add a pseudo-random value to this identifier to allow several CIFS messages to be attached to one SOAP message.

Reduced Mapping Mode

Content-type for attachments (i.e., file contents) must be application/octet-stream. In addition, the SOAP envelope must contain the *real* MIME type of the file being transferred to allow proper handling of the attachment on

the receiver end (e.g. determine which program should be used to open it). Unless this is decided based on the file extension (e.g. *.jpg etc.).

<AttachmentRealMIMEType>image/jpeg</AttachmentRealMIMEType>

<... Description of each element in the messages ...>

Administration interfaces

a. IADMListHosts()

The request message in this interface must conform to IADMListHostsRequest.xsd:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<xs:schema xmlns:xs="http://www.ongx.org/CIFS2XML">
      <xs:element name="IADMListHostsRequest"/>
</xs:schema>
```

A response must conform to IADMListHostsResponse.xsd:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
 <xs:schema xmlns:xs="http://www.ongx.org/CIFS2XML">
       <xs:element name="IADMListHostsResponse">
             <xs:complexType>
                   <xs:sequence>
                         <xs:element name="host" type="xs:string"
                         minOccurs="0" max="unbounded"/>
             </xs:sequence>
             </xs:complexType>
       </xs:element>
 </xs:schema>
```

b. IADMListUsers()

A request must conform to *IADMListUsersRequest.xsd*:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<xs:schema xmlns:xs="http://www.ongx.org/CIFS2XML">
      <xs:element name="IADMListUsersRequest"/>
```

```
</xs:schema>
```

A response must conform to *IADMListUsersResponse.xsd*:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>

<xs:schema xmlns:xs="http://www.ongx.org/CIFS2XML">
      <xs:element name="IADMListUsersResponse">
            <xs:complexType>
                  <xs:sequence>
                        <xs:element name="User" type="xs:string"
                        minOccurs="0" max="unbounded"/>
            </xs:sequence>
            </xs:complexType>
      </xs:element>
</xs:schema>
```

c. *IADMListDirectoriesOnHost(Host)*

A request must conform to *IADMListDirectoriesOnHostRequest.xsd*:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<xs:schema xmlns:xs="http://www.ongx.org/CIFS2XML">
      <xs:element name="IADMListDirectoriesOnHostRequest">
            <xs:element name="host" type="xs:string"/>
      </xs:element>
</xs:schema>
```

A response must conform to *IADMListSharesOnHostResponse.xsd*:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<xs:schema xmlns:xs="http://www.ongx.org/CIFS2XML">
      <xs:element name="IADMListDirectoriesOnHostResponse">
            <xs:complexType>
                  <xs:sequence>
                        <xs:element name="directory"
                        type="xs:string" minOccurs="0"
                        max="unbounded"/>
            </xs:sequence>
            </xs:complexType>
      </xs:element>
```

```
</xs:schema>
```

d. IADMSetAccessRights(URI resource, Int accessrights)

A request must conform to IADMSetAccessRightsRequest.xsd:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<xs:schema xmlns:xs="http://www.ongx.org/CIFS2XML">
      <xs:element name="IADMSetAccessRightsRequest">
            <xs:element name="resource" type="xs:anyURI"/>
            <xs:element name="accessrights" type="xs:int"/>
      </xs:element>
</xs:schema>
```

A response must conform to IADMSetAccessRightsResponse.xsd:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<xs:schema xmlns:xs="http://www.ongx.org/CIFS2XML">
      <xs:element name="IADMGetUserConfigurationResponse">
            <xs:element name="result" type="xs:boolean"/>
      </xs:element>
</xs:schema>
```

e. IADMGetUserConfiguration(User)

A request most conform to *IADMGetUserConfigurationRequest.xsd*:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<xs:schema xmlns:xs="http://www.ongx.org/CIFS2XML">
<xs:element name="IADMGetUserConfigurationRequest">
      <xs:element name="user_id" type="xs:string"/>
</xs:element>
</xs:schema>
```

A response must conform to *IADMGetUserConfigurationResponse.xsd*:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
      <xs:schema xmlns:xs="http://www.ongx.org/CIFS2XML">
      <xs:element name="IADMGetUserConfigurationResponse">
            <xs:element name="configuration" type="xs:configuration"/>
      </xs:element>
</xs:schema>
```

f. IADMSetUserConfiguration(String user_id, Configuration c)

A request most conform to *IADMSetUserConfigurationRequest.xsd*:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
      <xs:schema xmlns:xs="http://www.ongx.org/CIFS2XML">
      <xs:element name="IADMGetUserConfigurationRequest">
            <xs:complextype>
                  <xs:sequence>
                        <xs:element name="user_id" type="xs:string"/>
                              <xs:element name="configuration"
                              type="xs:configuration"/>
                  </xs:sequence>
            </xs:complextype>
      </xs:element>
</xs:schema>
```

A response must conform to *IADMGetUserConfigurationResponse.xsd*:

```
      <?xml version="1.0" encoding="ISO-8859-1" ?>
      <xs:schema xmlns:xs="http://www.ongx.org/CIFS2XML">
            <xs:element name="IADMSetUserConfigurationResponse">
                  <xs:element name="result" type="xs:boolean"/>
            </xs:element>
      </xs:schema>
```

Connection Establishment, Maintenance and Teardown interfaces

These functions are transparent to the Home Access Client, and will be performed only by the Home Access Local Web Service. The mapping is thus one-to-one between client calls and file system calls. This is already discussed above.

Resource Access interfaces

a. IACCListResources(URI uri, String pattern, Boolean recursive)

A request on this interface must conform to *IACCListResourcesRequest.xsd*:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<xs:schema xmlns:xs="http://www.ongx.org/CIFS2XML">
<xs:element name="IACCListResourcesRequest">
  <xs:complexType>
        <xs:sequence>
                  <xs:element name="uri" type="xs:anyURI"/>
                  <xs:element name="pattern"
                  type="xs:string"/>
                  <xs:element name="recursive"
                  type="xs:boolean"/>
        </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>
```

A response on this interface must conform to *IACCListResourcesResponse.xsd*:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<xs:schema xmlns:xs="http://www.ongx.org/CIFS2XML">
<xs:element name="IACCListResourcesResponse">
  <xs:complexType>
        <xs:sequence>
                  <xs:element name="resource"
                  type="xs:string" minOccurs="0"
                  max="unbounded"/>
        </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>
```

b. *IACCReadResource(URI uri)*

A request on this interface must conform to *IACCReadResourceRequest.xsd*:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<xs:schema xmlns:xs="http://www.ongx.org/CIFS2XML"
<xs:element name="IACCReadResourceRequest">
  <xs:complexType>
        <xs:sequence>
                  <xs:element name="uri" type="xs:anyURI"/>
        </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>
```

A response on this interface must conform to *IACCReadResourceResponse.xsd*:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<xs:schema xmlns:xs="http://www.ongx.org/CIFS2XML"
            xmlns:ref="http://ws-
      i.org/profiles/basic/1.1/xsd">
```

```
<xs:element name="IACCReadResourceResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="resource"
      type="ref:swaRef" use="required"/>
    </xs:sequence>
  </xs:complexType>
 </xs:element>
</xs:schema>
```

c. *IACCWriteResource(URI uri, WriteSpecification ws)*

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<xs:schema xmlns:xs="http://www.ongx.org/CIFS2XML">
<xs:element name="IACCWriteResourceRequest">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="uri" type="xs:anyURI"/>
      <xs:element name="writespecification"
      type="xs:writespecification"/>
    </xs:sequence>
  </xs:complexType>
 </xs:element>
</xs:schema>
```

A response on this interface must conform to *IACCListResourcesResponse.xsd*:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<xs:schema xmlns:xs="http://www.ongx.org/CIFS2XML">
<xs:element name="IACCWriteResourceResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="writeresult"
      type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
 </xs:element>
</xs:schema>
```

Information sources included by reference

[1] Sun Microsystems Inc. (1989). NFS: Network File System Protocol Specification. IETF. March 1989. http://www.ietf.org/rfc/rfc1094.txt?number=1094

[2] Callaghan, B., et.al. (1995). NFS Version 3 Protocol Specification. IETF. June 1995. http://www.ietf.org/rfc/rfc1813.txt?number=1813

[3] Storage Networking Industry Association (SNIA). (2002). Common Internet File System (CIFS) Technical Reference Revision 1.0. February 2002.

[4] Barton, J., et.al. (2000). SOAP Messages with Attachments. World Wide Web Consortium (W3C). December 2002. http://www.w3.org/TR/SOAP-attachments

[5] Gudgin M., et.al. (editors) (2005). SOAP Message Transmission Optimization Mechanism. World Wide Web Consortium (W3C). January 2005. http://www.w3.org/TR/2005/REC-soap12-mtom-20050125/

[6] Ferris, C., et.al. (editors) (2004). Attachments Profile Version 1.0. Web Services Interoperability Organization (WS-I). August 2004. http://www.ws-i.org/Profiles/AttachmentsProfile-1.0-2004-08-24.html

[7] Sun Microsystems, Inc. (2003). JSR-67: Java APIs for XML Messaging 1.0. Java Community Process (JCP). http://www.jcp.org/en/jsr/detail?id=67

[8] Sun Microsystems, Inc. (2003). JSR-101: Java APIs for XML based RPC. Java Community Process (JCP). http://www.jcp.org/en/jsr/detail?id=101

[9] Franks, J., et.al. (1999). HTTP Authentication: Basic and Digest Access Authentication. IETF. June 1999. http://www.ietf.org/rfc/rfc2617.txt?number=2617

Claims

1. A method for providing access to services and files on a computer (4) in a local network (1) from a stationary or mobile device (8, 9) outside said local network, said local network being equipped with a networked file system, and said stationary or mobile device (8, 9) being able to communicate with said local network over a wide area network,
characterized in that
for any networked file system message that is to be transmitted over said wide area network mapping at least some fields of said networked file system message into corresponding fields in an XML message representing said networked system message, and
for any XML message received over said wide area network parsing said XML message, and if said XML message represents a networked file system message reconstructing a networked file system message by mapping each field of said XML message into a corresponding field of said reconstructed networked file message.

2. A method as claimed in claim 1,
characterized in that said XML message being a SOAP message.

3. A method as claimed in claim 1,
characterized in said stationary or mobile device supporting a reduced set of XML format messages, and only the most important parts of the networked file system messages.

4. A method as claimed in claim 1,
characterized in mapping said networked file system messages into XML format messages using XML Schema Definitions.

5. A method as claimed in claim 1, characterized in providing file content in application/octet-stream format.

6. A method as claimed in claim 1, characterized in providing information regarding the MIME type of a file being transferred.

7. A device for providing access to services and files on a computer (4) in a local network (1), from a stationary or mobile device (8, 9) outside said local network, said local network being equipped with a networked file system, and said stationary or mobile device (8, 9) being connected to said local network (1) over a wide area network, characterized in that said device being adapted to map at least some fields of a networked file system message to be transmitted over said wide area network into corresponding fields in an XML message representing said networked file system message,
said device further being adapted to parse a XML message received over said wide area network, and if said XML message represents a networked file system message to reconstruct a networked file system message by mapping each field of said XML message into a corresponding field of said reconstructed networked file message.

8. A device as claimed in claim 7, characterized in that said device being installed in the local network (1) as a Home Access Local Web service (5).

9. A device as claimed in claim 8, characterized in that said device being installed on said stationary or mobile device (8, 9) as a Home Access Web Services Client (7), said Home Access Web Services Client (7) being adapted to interact with said Home Access Local Web service (5) to access files and services on the local network (1).

10. A device as claimed in claim 7, characterized in that said XML message being a SOAP message.

11. A device as claimed in claim 7, characterized in that said stationary or mobile device supporting a reduced set of XML format messages, and only the most important parts of the networked file system messages.

12. A device as claimed in claim 7, characterized in that XML Schema Definitions are used for mapping networked file system messages into XML format messages.

13. A device as claimed in claim 7, characterized in that said device being adapted to provide file content of application/octet-stream format.

14. A device as claimed in claim 7, characterized in that said device being adapted to provide information regarding the MIME type of a file being transferred.

15. A device as claimed in claim 7, characterized in that said networked file system being a Sun Network File System or a Common Internet File System.

16. A device as claimed in claim 9, characterized in a Home Access Global Web Service (6) addressable by a global IP address, said Home Access Global Web Service (6) being adapted to hold the current IP address of the local network (1).

17. A device as claimed in claim 16, characterized in the Home Access Global Web Service (6) being adapted to relay method requests from the

Home Access Web Services Client (7) to the Home Access Local Web Service (5).

18. A device as claimed in claim 16, characterized in that said Home Access Local Web Service (5) being adapted to communicate with the Home Access Global Web Service (6) and update its current IP address.

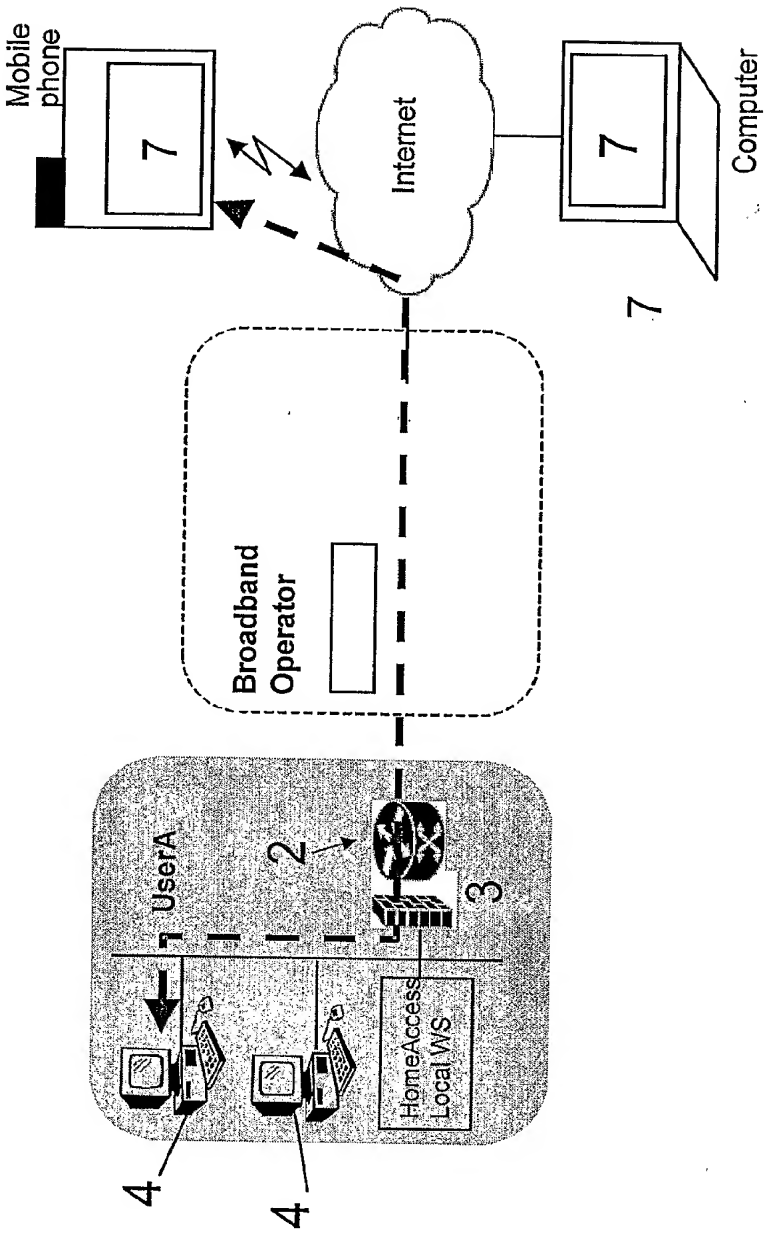

Mobile phone
7
Internet
7
Computer
7
Broadband Operator
UserA
2
3
HomeAccess
Local WS
4
4

Figure 1

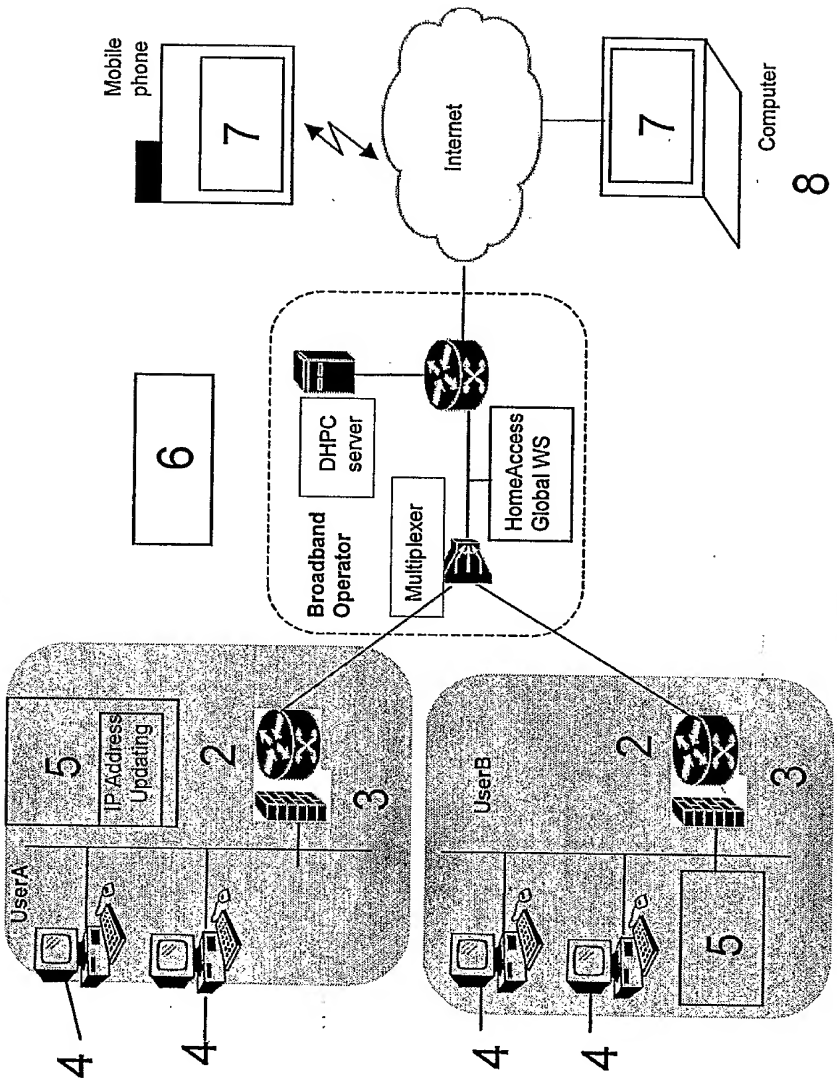

Mobile phone
7
Internet
7
Computer
8
6
DHPC server
HomeAccess Global WS
Broadband Operator
Multiplexer
5
IP Address Updating
2
3
UserA
UserB
4
4
4
4

Figure 2

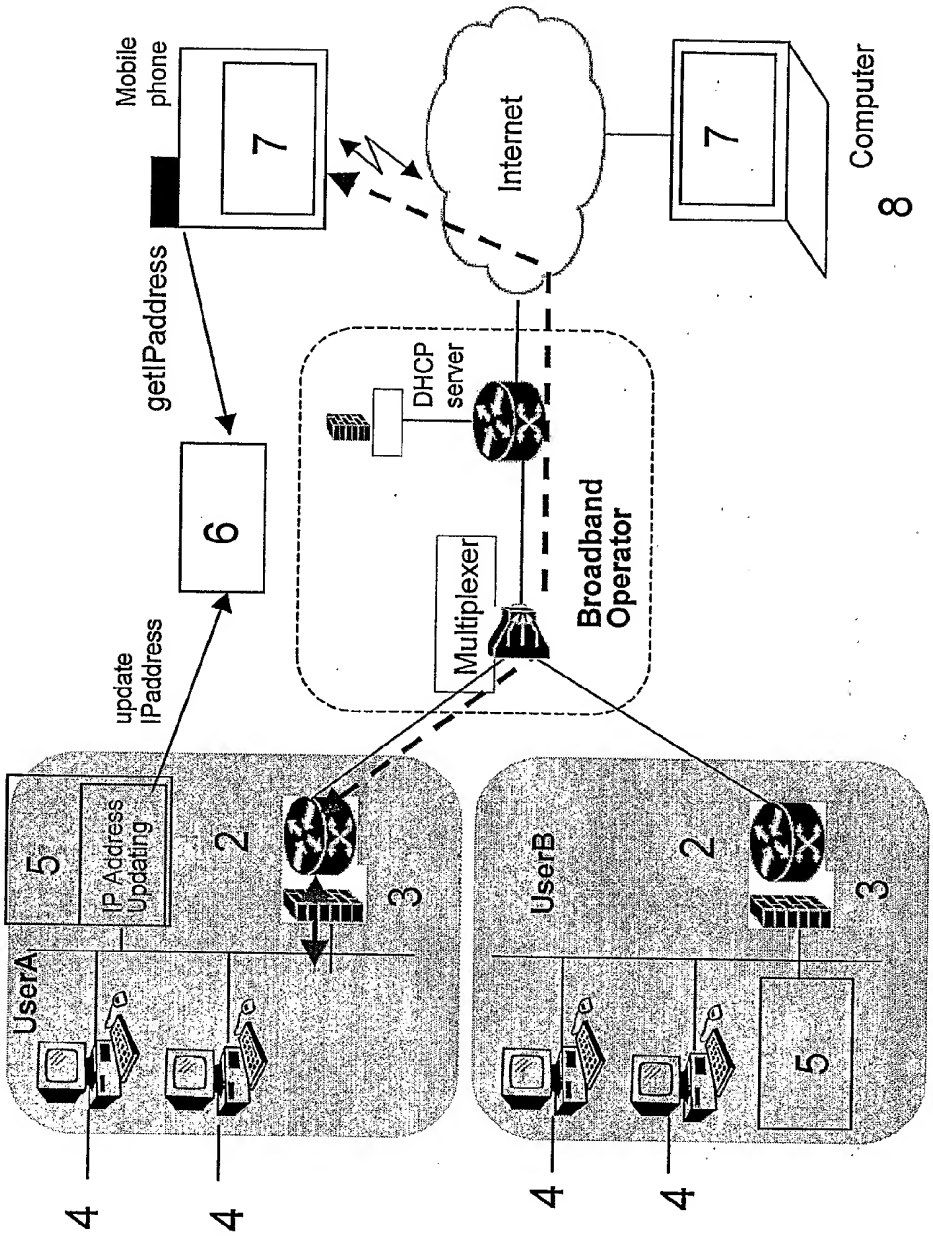

Mobile phone
7
Internet
7
Computer
8
getIPaddress
6
DHCP server
Broadband Operator
Multiplexer
update IPaddress
5
IP Address Updating
2
3
UserA
UserB
4
4
4
4

Figure 3

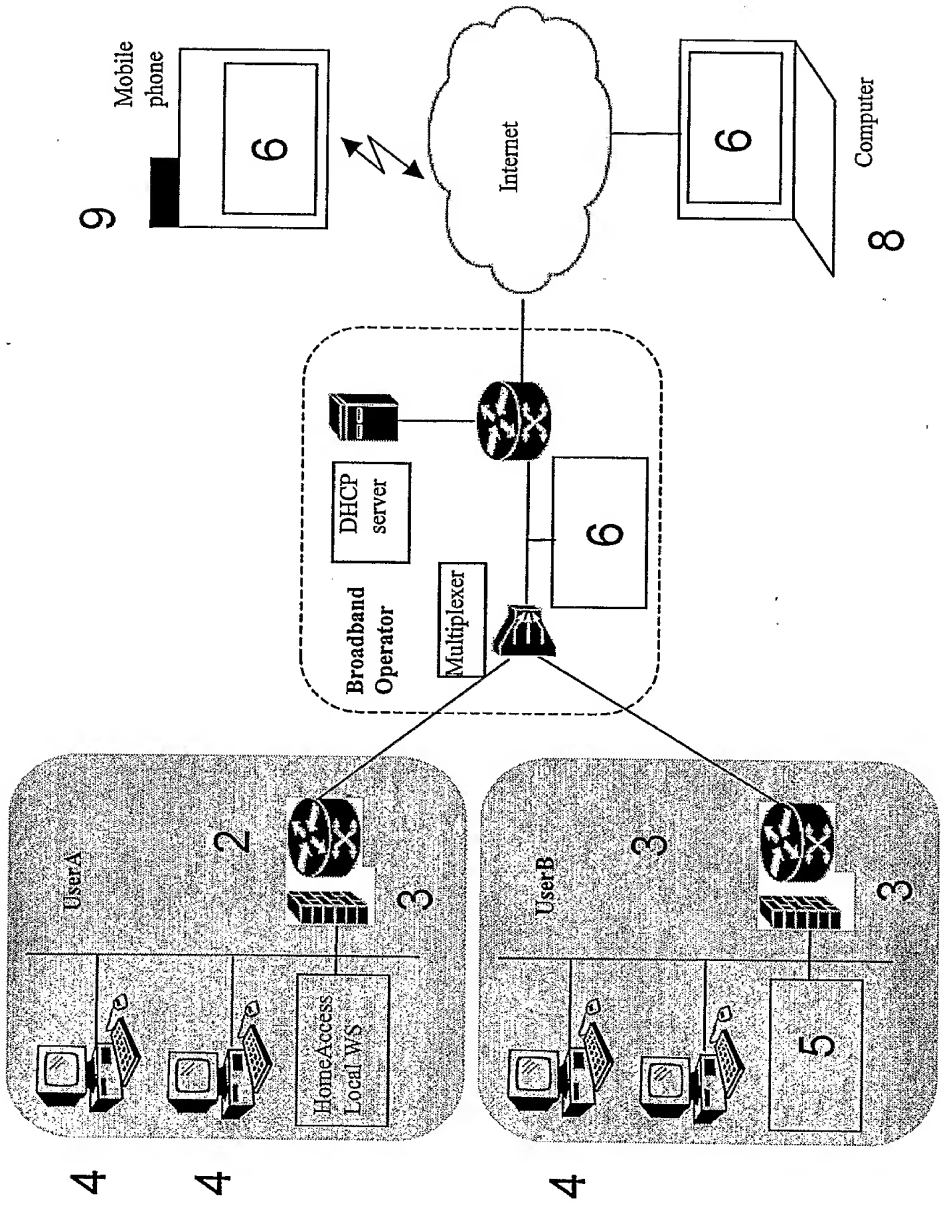

Mobile phone
9
6
Internet
6
Computer
8
DHCP server
Broadband Operator
Multiplexer
6
User A
2
3
HomeAccess Local WS
User B
3
3
5
4
4
4

Figure 4

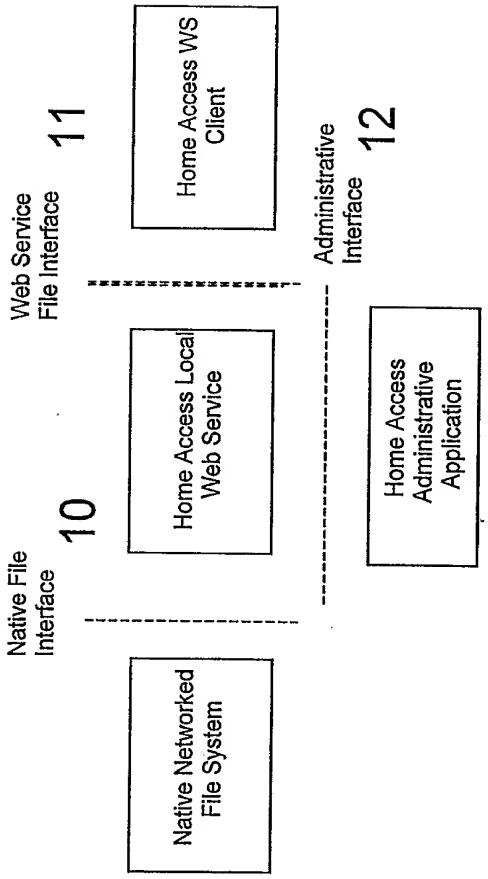

Native File Interface
10
Web Service File Interface
11
Administrative Interface
12
Native Networked File System
Home Access Local Web Service
Home Access Administrative Application
Home Access WS Client

Figure 5

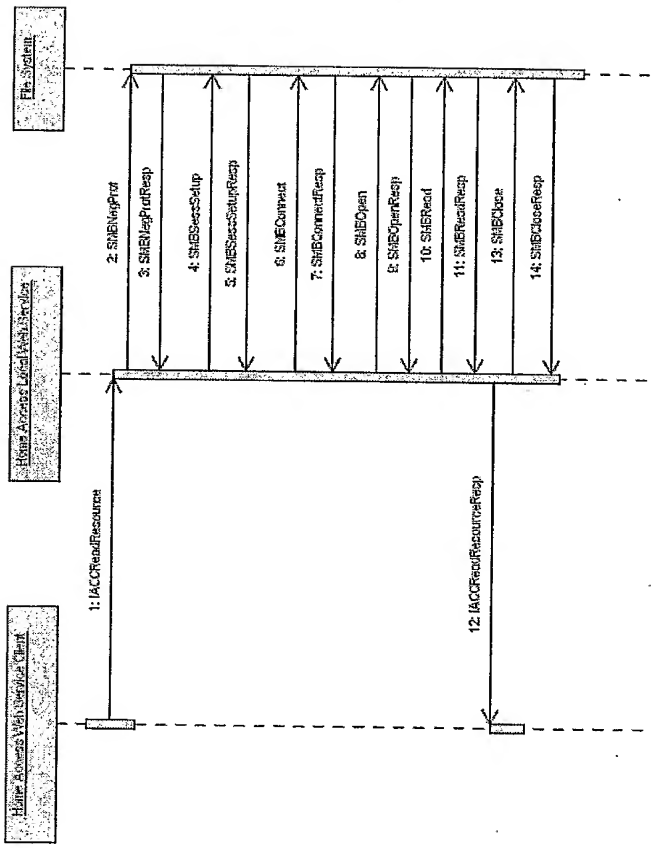

File System
Home Access Web Service
Home Access Web Service Client
1: IACCReadResource
2: SMBNegProt
3: SMBNegProtResp
4: SMBSessSetup
5: SMBSessSetupResp
6: SMBConnect
7: SMBConnectResp
8: SMBOpen
9: SMBOpenResp
10: SMBRead
11: SMBReadResp
12: IACCReadResourceResp
13: SMBClose
14: SMBCloseResp

Figure 6

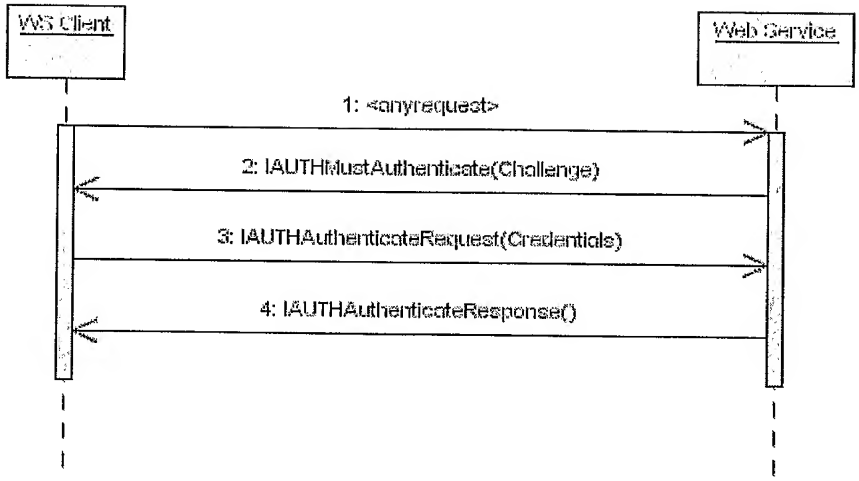

WS Client
Web Service
1: <anyrequest>
2: IAUTHMustAuthenticate(Challenge)
3: IAUTHAuthenticateRequest(Credentials)
4: IAUTHAuthenticateResponse()

Figure 7

INTERNATIONAL SEARCH REPORT

International application No
PCT/NO2006/000108

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04L29/06 H04L29/08

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal, PAJ, WPI Data, INSPEC

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2004/255048 A1 (LEV RAN ETAI ET AL) 16 December 2004 (2004-12-16) figures 1-4 paragraph [0243] - paragraph [0249] paragraph [0449] - paragraph [0450] paragraph [0465] - paragraph [0466] paragraph [0486] - paragraph [0488] -----	1-18
A	US 2002/032725 A1 (ARAUJO KENNETH S ET AL) 14 March 2002 (2002-03-14) figures 1,5-7,19 paragraph [0123] - paragraph [0126] paragraph [0129] - paragraph [0133] -----	1,7
	-/--	

[X] Further documents are listed in the continuation of Box C.

[X] See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
18 May 2006	01/06/2006

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL – 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Tyszka, K

Form PCT/ISA/210 (second sheet) (April 2005)

INTERNATIONAL SEARCH REPORT

International application No
PCT/NO2006/000108

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
P,X	DO VAN THANH, IVAR JORSTAD, DO VAN THUAN: "Fetching home music - Sending photos home" TELEKTRONIKK 3_4.2005, [Online] 2005, pages 123-130, XP002381495 Retrieved from the Internet: URL:http://www.telenor.com/telektronikk/volumes/index.php?page=ing&id1=67&id2=175&id3=879&select=05-09> [retrieved on 2006-05-18] the whole document	1-4, 7-12, 15-18

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/NO2006/000108

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2004255048 A1	16-12-2004	NONE	
US 2002032725 A1	14-03-2002	US 2001047406 A1	29-11-2001

DERWENT-ACC-NO: 2006-766876

DERWENT-WEEK: 200802

TITLE: Method of providing access to service and files on computer, involves reconstructing network file system message, by mapping each fields of extensible markup language message into corresponding field of network file message

INVENTOR: DO T V; JORSTAD I ; VAN THANH D ; VAN THUAN D

PATENT-ASSIGNEE: TELENOR ASA[TELEN]

PRIORITY-DATA: 2005NO-001487 (March 21, 2005)

PATENT-FAMILY:

PUB-NO	PUB-DATE	LANGUAGE
WO 2006101402 A1	September 28, 2006	EN
NO 200501487 A	September 22, 2006	NO
NO 323214 B1	January 29, 2007	NO
EP 1867128 A1	December 19, 2007	EN

DESIGNATED-STATES: AE AG AL AM AT AU AZ BA BB BG BR BW BY BZ CA CH CN CO CR CU CZ DE DK DM DZ EC EE EG ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KM KN KP KR KZ LC LK LR LS LT LU LV LY MA MD MG MK MN MW MX MZ NA NG NI NO NZ O M PG PH PL PT RO RU SC SD SE SG SK SL SM SY TJ TM TN TR TT TZ UA UG US UZ VC VN YU ZA ZM ZW AT BE BG BW CH CY CZ DE DK EA EE ES FI FR GB GH GM GR HU IE IS IT KE LS LT LU LV MC MW MZ NA NL OA PL PT RO SD SE SI SK SL SZ TR TZ UG ZM ZW AT BE BG CH CY CZ DE D K EE ES FI FR GB GR HU IE IS IT LI LT LU LV MC NL PL PT RO SE SI SK TR

APPLICATION-DATA:

PUB-NO	APPL-DESCRIPTOR	APPL-NO	APPL-DATE
WO2006101402A1	N/A	2006WO-NO000108	March 21, 2006
NO 200501487A	N/A	2005NO-001487	March 21, 2005
NO 323214B1	N/A	2005NO-001487	March 21, 2005
EP 1867128A1	N/A	2006EP-716777	March 21, 2006
EP 1867128A1	Based on	2006WO-NO000108	March 21, 2006

INT-CL-CURRENT:

TYPE	IPC DATE
CIPP	H04L12/66 20060101
CIPP	H04L12/66 20060101
CIPP	H04L29/06 20060101
CIPS	H04L29/08 20060101
CIPS	H04L29/10 20060101
CIPS	H04L29/10 20060101
CIPS	H04Q7/24 20060101
CIPS	H04Q7/24 20060101

ABSTRACTED-PUB-NO: WO 2006101402 A1

BASIC-ABSTRACT:

NOVELTY - The method involves mapping fields of network file system message into corresponding fields in an extensible markup language (XML) message representing the network system message, for any network file system message that is to be transmitted over wide area network. The XML message received over wide area network is parsed, and network file system message is reconstructed, by mapping each fields of XML message into corresponding field of reconstructed network file message, if XML message represents network file system message.

DESCRIPTION - An INDEPENDENT CLAIM is included for device for providing access to service and files on computer.

USE - For providing access to service and files on computer, from remote computer, personal digital assistant (PDA) and mobile phone.

ADVANTAGE - The method allows several computers on the local area network to provide services, thus every person having their own computer can access their personal services from their mobile device.

DESCRIPTION OF DRAWING(S) - The figure shows an explanatory

drawing of the local area network.

local area network (1)

broadband router (2)

firewall (3)

computers (4)

client (7)

CHOSEN-DRAWING: Dwg.2/7

TITLE-TERMS: METHOD ACCESS SERVICE FILE COMPUTER RECONSTRUCT NETWORK SYSTEM MESSAGE MAP FIELD EXTEND LANGUAGE CORRESPOND

DERWENT-CLASS: T01

EPI-CODES: T01-N02B1;

SECONDARY-ACC-NO:

Non-CPI Secondary Accession Numbers: 2006-594164